HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P.O. Box 272400
Fort Collins, Colorado 80527-2400

PATENT APPLICATION

ATTORNEY DOCKET NO. 200314241-1

IN THE
UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor(s): Jonathan Paul PATRIZIO et al.

Confirmation No.: 5514

Application No.: 10/767,524

Examiner: Eun Hee CHUNG

Filing Date: 2004-Jan-29

Group Art Unit:

Title: FAILURE-RESPONSE SIMULATOR FOR COMPUTER CLUSTERS

Mail Stop Appeal Brief-Patents
Commissioner For Patents
PO Box 1450
Alexandria, VA 22313-1450

TRANSMITTAL OF APPEAL BRIEF

Transmitted herewith is the Appeal Brief in this application with respect to the Notice of Appeal filed on 2008-Jun-11.

☒ The fee for filing this Appeal Brief is $540.00 (37 CFR 41.20) less $510 credit.

☐ No Additional Fee Required.

(complete (a) or (b) as applicable)

The proceedings herein are for a patent application and the provisions of 37 CFR 1.136(a) apply.

☐ (a) Applicant petitions for an extension of time under 37 CFR 1.136 (fees: 37 CFR 1.17(a)-(d)) for the total number of months checked below:

☐ 1st Month $130	☐ 2nd Month $490	☐ 3rd Month $1110	☐ 4th Month $1730

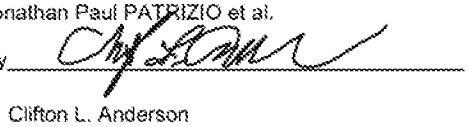
☐ The extension fee has already been filed in this application.

☒ (b) Applicant believes that no extension of time is required. However, this conditional petition is being made to provide for the possibility that applicant has inadvertently overlooked the need for a petition and fee for extension of time.

Please charge to Deposit Account 08-2025 the sum of $30. At any time during the pendency of this application, please charge any fees required or credit any over payment to Deposit Account 08-2025 pursuant to 37 CFR 1.25. Additionally please charge any fees to Deposit Account 08-2025 under 37 CFR 1.16 through 1.21 inclusive, and any other sections in Title 37 of the Code of Federal Regulations that may regulate fees.

Respectfully submitted,

Jonathan Paul PATRIZIO et al.

By 

Clifton L. Anderson

Attorney/Agent for Applicant(s)

Reg No. : 30,989

Date : 2008-Nov-14

Telephone : (408) 257-6070

H0312-8BE

Before The Board of Patent Appeals & Interferences

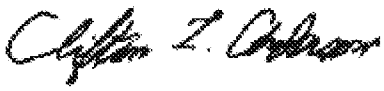
For: FAILURE-RESPONSE SIMULATOR FOR COMPUTER CLUSTERS			
Applicant: Jonathan Paul PATRIZIO et al.		Attorney Docket No.: 200314241-1	
Serial No.: 10/767,524 (5514)	Filed: January 29, 2004	Art Unit: 2123	Examiner: Eun Hee CHUNG

Director For Patents
PO Box 1450
Alexandria, VA 22313-1450

Appeal Brief (Identification Page)

This is an appeal to the Board of Patent Appeals and Interferences from the Office Action mailed 2008-Aug-21 in the above-identified patent application. The Office Action reopened prosecution in response to an Appeal Brief filed 2008-Jun-11. A Notice of Appeal for the present Appeal Brief is being filed herewith on 2008-Nov-14.

Respectfully submitted
Jonathan Paul PATRIZIO et al.
by



Clifton L. Anderson
Reg. No. 30,989
(408) 257-6070

TABLE OF CONTENTS

REAL PARTY IN INTEREST

The real parties in interest are

Hewlett-Packard Company, a Delaware corporation; and

Hewlett-Packard Development Company, L.P., a Texas limited partnership and wholly owned affiliate of Hewlett-Packard Company, and assignee of record of the Appellants' rights.

RELATED APPEALS AND INTERFERENCES

None.

There are no related appeals or interferences.

STATUS OF CLAIMS

Claims 1-16 are pending in the application.

Claims 1-16 are rejected.

The rejections of Claims 1-16 are being appealed.

STATUS OF AMENDMENTS

All amendments have been entered. There are no unentered amendments.

SUMMARY OF CLAIMED SUBJECT MATTER

Summary Overview

Some computer clusters can reconfigure themselves to continue a mission despite a failure that would halt the mission in the absence of such reconfiguration. The present invention provides a failure-response simulator for evaluating the failure-response of a cluster configuration. The failure-response simulator accepts pre-failure configurations and virtual failure events as inputs and, in response, outputs virtual post-failure configurations. The failure-response simulator can be contrasted with a performance simulator that accepts a configuration and load parameters as inputs and, in response, outputs performance data.

Concise Explanation of Claim 1

Claim 1 relates to a computer system (AP1, Fig. 1, paragraph 26, page 5, lines 11-13) comprising:

a simulator (SIM, Fig. 1, paragraph 27, page 5, line 18 to page 6, line 2) including:

a virtual-failure event selector (V06, Fig. 5, paragraph 39, page 9, lines 26-27) providing for selecting a virtual-failure event corresponding to a real-failure event that applies to a real computer cluster (RCC, Fig. 1), and

a virtual-cluster generator (V04) for generating a first virtual cluster (VC1, Fig. 6) in a virtual pre-failure configuration corresponding to a real pre-failure configuration of said real computer cluster, and for, in response to selection of said virtual-failure event, generating a second virtual cluster (VC2, Fig. 7) in a virtual post-failure configuration corresponding to a real post-failure configuration that said real computer cluster would assume in response to said real-failure event.

Concise Explanation of Claim 10

Claim 10 relates to a computer-implemented method comprising:

a) generating (S4, Fig. 6, paragraph 45, page 12, lines 6-10) a first virtual computer cluster (VC1, Fig. 1) in a virtual pre-failure configuration that serves as a model for a real computer cluster (RCC, Fig. 1) in a pre-failure configuration that responds to predetermined types of failures by reconfiguring to a real post-failure configuration, said reconfiguring including migrating a real application on one real computer of said real computer cluster to another real computer of said real computer cluster;

b) selecting (S5, Fig. 6, paragraph 45, page 12, lines 7-12) a sequence of at least one of said predetermined types of failures; and

c) generating (S6, Fig. 6, paragraph 45, page 12, lines 8-10) a second virtual computer cluster (VC2, Fig. 7) in a virtual post-failure configuration that serves as a model for said real computer cluster in said real post-failure configuration.

GROUNDS OF REJECTION TO BE REVIEWED

To be reviewed are the rejections of Claims 1-16 under 35 U.S.C. 103(a) for being unpatentable over U.S. Patent No. 7,107,191 to Stewart et al., "Stewart" herein, in view of U.S. Patent No. 7,228,458 to Kesavan, "Kesavan" herein.

ARGUMENTS

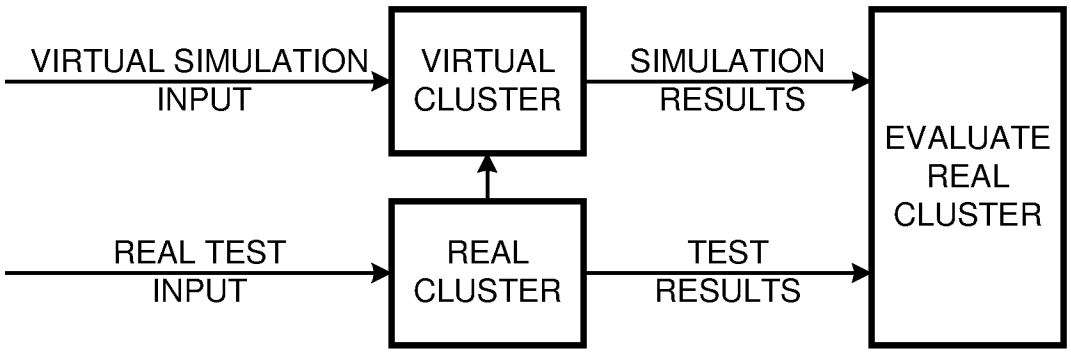
[01] Arguments for reversing the rejections of Claims 1-16 under 35 U.S.C. 103(a) for being unpatentable over Stewart in view of Kesavan.

[02] For the purposes of these arguments, the claims are divided into two groups: Group 1 includes independent Claim 1 and its dependents Claims 2-9; and Group 2 includes independent Claim 10 and its dependents Claims 11-16.

[03] OVERVIEW: Testing vs. Simulation

[04] The Office Action relies on Kesavan for a disclosure of a second virtual cluster, but Kesavan does not disclose a virtual cluster, second or otherwise. Kesavan discloses *testing of a real cluster*, rather than *simulation* of a real cluster *using a virtual cluster*. Since Kesavan involves testing rather than simulation of a cluster, Kesavan does not need to and in fact does not disclose a virtual cluster. Accordingly, the claimed second virtual cluster is not disclosed either by Stewart (as the Office Action admits) or by Kesavan (as the Office Action erroneously concludes). Accordingly, any modification of Stewart in accordance with the teachings of Kesavan would fail to meet the claim limitations. Accordingly, the rejections for obviousness should be reversed.

[05] The distinction between testing and simulation is represented in the following figure.



In either case, the goal is to evaluate a real cluster. Testing facilitates this evaluation by applying test inputs to the real cluster and observing the results in the form of an impact on the real system. Simulation facilitates this evaluation by generating a virtual cluster corresponding to the real cluster, applying virtual inputs to the virtual cluster and observing the effects on the virtual cluster. The most obvious difference is that testing does not require generation of a virtual cluster. Kesavan, which involves testing, does not disclose generation of a virtual cluster as called for in the claims.

[06] GROUP 1: CLAIMS 1-10

[07] 2143.03 All Claim Limitations Must Be Taught or Suggested

[08] To establish *prima facie* obviousness of a claimed invention, all the claim limitations must be taught or suggested by the prior art. *In re Royka*, 490 F.2d 981, 180 USPQ 580 (CCPA 1974). "All words in a claim must be considered in judging the patentability of that claim against the prior art." *In re Wilson*, 424 F.2d 1382, 1385, 165 USPQ 494, 496 (CCPA 1970). If an independent claim is nonobvious under 35 U.S.C. 103, then any claim depending therefrom is nonobvious. *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988). (MPEP, 2100-131 Rev. 5, Aug. 2006)

[09] The Office Action relies on Kesavan for a disclosure of the following claim elements that it recognizes are not taught by Stewart.

> a virtual-cluster generator for, in response to selection of said virtual-failure event, generating a second virtual cluster in a virtual post-failure configuration corresponding to a real post-failure configuration that said real computer cluster would assume in response to said real-failure event (Col. 7 lines 11-24, Fig 3-4, Col. 2 lines 54-67, Col. 3lines 1-3, Col. 4 lines 36-39, Col. 5 lines 12-17, Col. 5 lines 57-67, col. 6 lines 1-16, Col. 6 lines 40-50,).

[10] However, Kesavan does not teach or suggest: 1) a virtual-cluster generator, 2) a virtual-failure event, 3) a second (or a first) virtual cluster, or 4) a virtual post-failure configuration. A failure to disclose even one of these elements (that are admittedly not disclosed by Stewart) would be sufficient justification for reversing the rejections for obviousness.

[11] The foregoing passage from the Office Action refers to Kesavan, Figs. 3 and 4. Kesavan discloses (column 1, lines 66-67) that "FIG. 3 is a block diagram of a clustered system having a failed node, according to one embodiment of the invention." Kesavan discloses (column 2, lines 1-3) that "FIG. 4 is a block diagram of a clustered system having a failed connection between two nodes, according to one embodiment of the invention." Thus, Kesevan, Figs. 3 and 4 depict *real* (not virtual) clustered systems in *real* post-failure configurations. Figs. 3 and 4 do not depict a virtual cluster in a virtual-post failure configuration.

[12] The foregoing passage from the Office Action refers to several textual passages in Kesavan. Not one of these refers to a virtual cluster, either in those terms or in any other. Thus, the Office Action fails to establish that Kesavan discloses the claimed second virtual cluster. The Office Action concedes that Stewart does not disclose the claimed second virtual cluster. Since the Office Action fails to establish that the cited art discloses the claimed second virtual cluster, the rejections for obviousness of Claims 1-10 should be withdrawn.

[13] Of course the reason that the Office Action fails to establish that Kevasan teaches or suggests a virtual cluster is that Kevasan does not, in fact, disclose a virtual cluster. Kevasan discloses testing rather than simulation. The computing nodes and storage devices involved in the testing are real, not virtual.

[14] Appellant agrees that it would be obvious to modify Stewart in accordance with the teachings of Kevasan. This could be done by installing Kevasan's test programs on Stewart's cluster system. This system would still be able to simulate the performance of possible configurations, and would add the capability of qualifying storage

devices as taught by Kevasan. This simple and straightforward combination would achieve the advantages of both references.

[15] However, the combination proposed by the Office Action is not the obvious one. The Office Action appears to propose modifying Stewart's performance simulator to accept test inputs that simulate failure scenarios. However, nothing in Stewart suggests that the performance simulator is designed to accept such inputs. Furthermore, nothing in either reference would suggest that Stewart's simulator would respond to such inputs by generating a second virtual cluster in a second virtual configuration. Thus, while there is an obvious way to combine Stewart and Kevasan, there is no obvious way to modify Stewart's performance simulator in accordance with the Kevasan's teachings.

[16] Thus, there is no motivation or rationale for the modification of Stewart in accordance with Kesavan as proposed in the Office Action. Even if the proposed combination were obvious, it would still not meet all claim limitations, e.g., the proposed combination would not meet the requirement for a virtual-cluster generator that generates the claimed second virtual cluster. **Since the proposed combination is not obvious, and since the proposed combination would not meet all the limitations of Claim 1, the rejections for obviousness of Claims 1-10 should be reversed.**

[17] GROUP 2: CLAIMS 10-16

[18] The arguments for reversing the rejections for obviousness of Claims 11-16 are basically the same as those for Claims 1-10. While it is obvious to modify a system as disclosed by Stewart by including the test software taught by Kesavan to obtain the advantages of both, there is no motivation for modifying Stewart's simulator so that it can accept selections of tests that simulate failure events. One skilled in the art would not see any benefit to such a modification, which would appear unfeasible since Stewart does not teach that the simulator can handle such inputs.

[19] However, even if the proposed modification were viable, it would not meet the limitations of Claim 10. Claim 10 requires "generating a second virtual computer cluster in a virtual post-failure configuration that serves as a model for said real computer cluster in said real post-failure configuration." This limitation is not disclosed by either of the cited references.

[20] In particular, neither reference discloses a second virtual computer cluster. The Office Action recognizes that Stewart does not disclose this limitation, but erroneously asserts that Kesavan discloses this limitation. However, Kesavan nowhere discloses a virtual cluster, let alone the second virtual computer cluster in a virtual post-failure configuration. . ." In any event, the Office Action has failed to identify which element disclosed by Kesavan corresponds to the claimed second virtual computer cluster and thus has failed to establish modifying Stewart in accordance with the teachings of Kesavan would meet the limitations of Claim 10.

[21] Since the proposed modification of Stewart according to Kesavan is not obvious and since the result would not meet the limitations of independent Claim 10, the rejections of Claims 10-16 for obviousness should be reversed.

CLAIMS APPENDIX

1. *(previously presented)* A computer system comprising:

a simulator including:

a virtual-failure event selector providing for selecting a virtual-failure event corresponding to a real-failure event that applies to a real computer cluster, and

a virtual-cluster generator for generating a first virtual cluster in a virtual pre-failure configuration corresponding to a real pre-failure configuration of said real computer cluster, and for, in response to selection of said virtual-failure event, generating a second virtual cluster in a virtual post-failure configuration corresponding to a real post-failure configuration that said real computer cluster would assume in response to said real-failure event.

2. *(previously presented)* A system as recited in Claim 1 wherein, in said real pre-failure configuration, said real computer cluster runs a software application on a first computer of said real computer cluster and not on a second computer of said real computer cluster, and wherein, in said real post-failure configuration, said real computer cluster runs said application on said second computer but not on said first computer.

3. *(original)* A system as recited in Claim 1 further comprising said real computer cluster, said real computer cluster including profiling software for providing a descriptive profile of said real computer cluster, said virtual-cluster generator generating said virtual cluster in said pre-failure configuration using said descriptive profile.

4. *(original)* A system as recited in Claim 3 wherein said real computer cluster is connected to said simulator for providing said descriptive profile thereto.

5. *(original)* A system as recited in Claim 2 wherein said simulator further includes an evaluator for evaluating said virtual cluster in its post-failure configuration.

6. *(original)* A system as recited in Claim 5 wherein said simulator further includes a test sequencer, said test sequencer selecting different virtual-failure events to be applied to said first virtual cluster in said pre-failure configuration so as to result in different post-failure configurations of said virtual cluster.

7. *(original)* A system as recited in Claim 6 wherein said simulator further includes a statistical analyzer for statistically analyzing evaluations of said different post-failure configurations of said virtual cluster.

8. *(original)* A system as recited in Claim 7 wherein said test sequencer automatically tests different pre-failure configurations of said virtual cluster against different failure events, said statistical analyzer providing a determination of optimum pre-failure configuration by statistically analyzing evaluations of the resulting post-failure configurations.

9. *(original)* A system as recited in Claim 8 wherein said simulator is connected to said real computer cluster for providing said determination thereto, said real computer cluster automatically reconfiguring itself as a function of said determination.

10. *(previously presented)* A computer-implemented method comprising:

a) generating a first virtual computer cluster in a virtual pre-failure configuration that serves as a model for a real computer cluster in a pre-failure configuration that responds to predetermined types of failures by reconfiguring to a real post-failure configuration, said reconfiguring including migrating a real application on one real computer of said real computer cluster to another real computer of said real computer cluster;

b) selecting a sequence of at least one of said predetermined types of failures; and

c) generating a second virtual computer cluster in a virtual post-failure configuration that serves as a model for said real computer cluster in said real post-failure configuration.

11. *(original)* A method as recited in Claim 10 wherein steps a, b, and c are iterated for different configurations of said real computer cluster and for different sets of said predetermined failure types, said method further comprising providing a recommended configuration for said real computer cluster.

12. *(original)* A method as recited in Claim 10 further comprising:

gathering profile information about said real cluster in said first configuration, wherein said first virtual computer cluster is generated using said profile information.

13. *(original)* A method as recited in Claim 12 wherein steps a, b, and c are iterated for different configurations of said real computer cluster and for different sets of said predetermined failure types, said method further comprising providing a recommended configuration for said real computer cluster.

14. *(original)* A method as recited in Claim 13 further comprising:

transmitting said recommendation to said real computer cluster; and

implementing said recommended configuration on said real computer cluster.

15. *(previously presented)* A method as recited in Claim 10 wherein said type of failure relates to a failure of a network interface or a hard disk interface.

16. *(previously presented)* A method as recited in Claim 1 wherein said real failure event involves a failure of a network interface or a hard disk interface.

EVIDENCE APPENDIX

No evidence is being submitted with this Appeal Brief.

RELATED PROCEEDINGS APPENDIX

None. There are no related proceedings.